STPR: SPATIOTEMPORAL PRESERVATION AND ROUTING FOR EXEMPLAR-FREE VIDEO CLASS-INCREMENTAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Video Class-Incremental Learning (VCIL) seeks to develop models that continuously learn new action categories over time without forgetting previously acquired knowledge. Unlike traditional Class-Incremental Learning (CIL), VCIL introduces the added complexity of spatiotemporal structures, making it particularly challenging to mitigate catastrophic forgetting while effectively capturing both frame-shared semantics and temporal dynamics. Existing approaches either rely on exemplar rehearsal, raising concerns over memory and privacy, or adapt static image-based methods that neglect temporal modeling. To address these limitations, we propose Spatiotemporal Preservation and Routing (StPR) mechanism, a unified and exemplar-free VCIL framework that explicitly disentangles and preserves spatiotemporal information. We begin by introducing Frame-Shared Semantics Distillation (FSSD), which identifies semantically stable and meaningful channels by jointly considering channel-wise sensitivity and classification contribution. By selectively regularizing these important semantic channels, FSSD preserves prior knowledge while allowing for adaptation. Building on this preserved semantic space, we further design a Temporal Decomposition-based Mixture-of-Experts (TD-MoE), which dynamically routes task-specific experts according to temporal dynamics, thereby enabling inference without task IDs or stored exemplars. Through the synergy of FSSD and TD-MoE, StPR progressively leverages spatial semantics and temporal dynamics, culminating in a unified, exemplar-free VCIL framework. Extensive experiments on UCF101, HMDB51, SSv2 and Kinetics400 show that our method outperforms existing baselines while offering improved interpretability and efficiency in VCIL. Code is available in the suppl. materials.

1 INTRODUCTION

Class-Incremental Learning (CIL) Li & Hoiem (2017); Belouadah et al. (2021); De Lange et al. (2021); Masana et al. (2022); Zhang et al. (2024) develops models that learn from a sequence of tasks without forgetting previous knowledge, recognizing an ever-growing set of classes without past task data or identifiers. A key challenge is catastrophic forgetting McCloskey & Cohen (1989); Ratcliff (1990), where new knowledge overwrites old. While well studied for images, extending CIL to videos: Video Class-Incremental Learning (VCIL) Park et al. (2021); Villa et al. (2022), remains underexplored. VCIL differs from CIL by requiring continual learning of new categories while modeling frame-shared semantics and temporal dependencies, unlike CIL's focus on static images. This spatiotemporal complexity is critical for understanding actions, motion, and scene dynamics in real-world applications like surveillance, driver monitoring, and robotics. Further, memory and privacy constraints often prohibit storing past data, demanding continual learning without rehearsal.

The central challenge of VCIL lies in ***mitigating catastrophic forgetting while effectively leveraging frame-shared semantics and temporal dynamics to incrementally learn new categories.*** Existing methods can be broadly categorized into two types, as illustrated in Figure 1(a): 1) Exemplar-based methods Rebuffi et al. (2017); Hou et al. (2019); Douillard et al. (2020); Park et al. (2021); Pei et al. (2022); Villa et al. (2022); Alssum et al. (2023); Liang et al. (2024); Chen et al. (2025) store a portion of previous data (video clips, frames, or features) and apply rehearsal to reduce forgetting. However, storing exemplars incurs memory and privacy costs and typically emphasizes

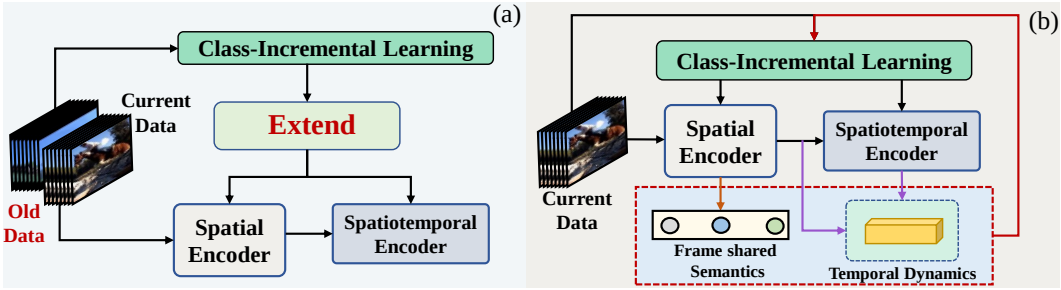

Figure 1: (a) Prior methods rely on exemplar rehearsal or naively stack video and CIL modules. (b) Our StPR framework explicitly decouples and reuses spatiotemporal semantics to mitigate forgetting.

frame-level learning without explicitly modeling temporal dynamics. 2) CIL-based methods Li & Hoiem (2017); Dhar et al. (2019); Cheng et al. (2024) adapt algorithms developed for static images, using techniques like regularization or subspace projection. While avoiding exemplar storage, they often overlook spatiotemporal properties by flattening or underutilizing temporal features. In contrast, our method StPR (Figure 1(b)) explicitly decouples video features into Frame shared semantics and temporal dynamics, and reuses these decomposed components to enhance the model's ability to adapt continually, thereby reducing forgetting without storing extensive exemplars.

Specifically, we propose a unified, exemplar-free VCIL framework that fully exploits the spatiotemporal nature of videos. Our method integrates both spatial semantic consistency and temporal variation to mitigate forgetting and improve adaptation across tasks. Separately, we introduce: 1) **Frame-Shared Semantics Distillation (FSSD).** To preserve frame-shared semantics and alleviate forgetting, we quantify the semantic importance of each channel using a combination of semantic sensitivity and classification contribution. This ensures that semantically meaningful and stable channels are preserved, achieving a better trade-off between stability and plasticity. 2) **Temporal-Decomposition-based Mixture-of-Experts (TD-MoE).** To exploit temporal dynamics for continual adaptation, we decouple task-specific temporal cues for each expert. At inference, expert routing depends solely on the temporal dynamics of the input, without requiring task identities or stored exemplars. This enables dynamic assignment of weights to experts according to the temporal dynamics of the input, facilitating incremental learning of new categories.

Our framework uniquely bridges the gap between video-specific spatiotemporal representation and class-incremental adaptation. By disentangling and leveraging both spatial semantic channel consistency and temporal dynamics, it offers an effective and explainable solution for continual video understanding. Our main contributions are: 1) We propose a Frame-Shared Semantics Distillation method (FSSD) that preserves frame-shared, semantically aligned spatial channels through semantic importance-aware regularization, optimizing the stability-plasticity trade-off in continual learningl; 2) We design a Temporal Decomposition based Mixture-of-Experts strategy (TD-MoE) that decomposes spatiotemporal features and uses temporal dynamics for expert combination, enabling task-id-free and dynamic adaptation; 3) We present a unified, exemplar-free VCIL framework that achieves state-of-the-art results on UCF101, HMDB51, SSv2 and Kinetics400, demonstrating the effectiveness of integrating spatial semantics and temporal dynamics in VCIL.

2 Related Work

2.1 Class-Incremental Learning

Class-Incremental Learning (CIL) aims to enable models to continually learn new classes without forgetting previously learned ones. Existing approaches typically fall into three categories: (1) *regularization-based methods* Kirkpatrick et al. (2017); Zenke et al. (2017); Xiang et al. (2022); Zhou et al. (2023), which constrain parameter updates to preserve prior knowledge, often via knowledge distillation Li & Hoiem (2017); Hou et al. (2019); (2) *exemplar-based methods* Bang et al. (2021); Chaudhry et al. (2018); Rebuffi et al. (2017), which store or generate past data to reduce forgetting; and (3) *structure-based methods* Serra et al. (2018); Mallya & Lazebnik (2018); Mallya et al. (2018); Liang & Li (2024); Yu et al. (2024), which expand model capacity or isolate task-specific components. Recently, CIL combined with pre-trained vision transformers (ViTs) Ermis et al. (2022); Smith et al. (2023); Wang et al. (2022b;c) has achieved impressive results by leveraging transferable representations and modularity. Some methods fully fine-tune pre-trained backbones Boschini et al.

(2022); Zhang et al. (2023), but this is computationally expensive. To address efficiency, parameter-efficient fine-tuning (PEFT) methods have been introduced. Prompt pool-based approaches Wang et al. (2022c); Smith et al. (2023); Wang et al. (2024); Zhang et al. (2023) maintain task-specific prompts, while adapter-based methods Zhou et al. (2024a); Tan et al. (2024); Gao et al. (2024); Liang & Li (2024); Zhou et al. (2024b) adapt ViTs to new classes with minimal updates. While effective, most CIL strategies were originally developed for static image domains and do not generalize well to video-based scenarios, where temporal dynamics play a critical role.

2.2 Video Class-Incremental Learning

Action recognition has been widely explored with 2D CNNs using temporal aggregation Lin et al. (2019); Wang et al. (2016) and 3D CNNs for joint spatiotemporal modeling Carreira & Zisserman (2017); Tran et al. (2015). More recent work focuses on improving temporal sensitivity and efficiency Feichtenhofer (2020); Fan et al. (2020). However, these models are trained in static setups and do not address continual adaptation or forgetting. To address these challenges, Video Class-Incremental Learning (VCIL) extends conventional Class-Incremental Learning (CIL) to spatiotemporal data, introducing additional challenges such as managing temporal variations across tasks. Several recent methods, including TCD Park et al. (2021), FrameMaker Pei et al. (2022), and HCE Liang et al. (2024), address this setting by storing videos or compressed exemplars. However, these strategies raise concerns related to memory efficiency and data privacy. While SMILE Alssum et al. (2023) effectively extracts image features from individual frames, it does not explicitly capture temporal information, which may limit its ability to leverage the distinctive decision cues present in video data. Exemplar-free methods such as STSP Cheng et al. (2024) mitigate forgetting via orthogonal subspace projections, but they mainly adapt image-domain strategies to video tasks. In contrast, our approach decouples and models the spatiotemporal structure of videos, proposing a unified VCIL framework that preserves spatial consistency via Frame-Shared Semantics Distillation (FSSD) for knowledge retention without exemplars, while leveraging temporal dynamics for expert routing through Temporal Decomposition-based Mixture-of-Experts (TD-MoE).

3 Method

Problem Definition: In the Video Class-Incremental Learning (VCIL) setting, a model is trained across B stages with sequentially arriving datasets $\{\mathcal{D}^1, \ldots, \mathcal{D}^B\}$. Each dataset $\mathcal{D}^b = \{(V_j^b, y_j^b)\}_{j=1}^{|\mathcal{D}^b|}$ corresponds to the b-th task, where V_j^b is the j-th video and y_j^b is its class label. Here, videos primarily represent human action recognition scenarios, where the spatiotemporal dynamics capture motion patterns, subject interactions, and scene context. $|\mathcal{D}^b|$ represents the number of samples in the b-th task. Let $\mathcal{Y}^b$ be the label space of the b-th dataset. For all $b \neq b'$, the label spaces are disjoint: $\mathcal{Y}^b \cap \mathcal{Y}^{b'} = \varnothing$. The objective of VCIL is to incrementally train a model over B tasks while maintaining high performance across all accumulated classes $\{\mathcal{Y}^1, \mathcal{Y}^2, \ldots, \mathcal{Y}^B\}$.

Overall framework. We propose a unified, exemplar-free framework for Video Class-Incremental Learning (VCIL) built upon the CLIP model Radford et al. (2021). Our goal is to mitigate catastrophic forgetting while effectively leveraging frame-shared semantics and temporal dynamics to incrementally learn new categories. The frozen visual encoder $\mathcal{F}(\cdot)$ extracts spatial features, while adapters $\mathcal{A}^b$ are updated for each task b. A spatiotemporal encoder $\mathcal{G}(\cdot)$ models temporal dynamics. Our framework introduces two key components: 1) Frame-Shared Semantics Distillation (FSSD) identifies semantically stable channels across frames by combining Semantic Sensitivity and Classification Score, applying selective regularization to preserve critical spatial semantics while maintaining plasticity. 2) Temporal Decomposition based Mixture-of-Experts (TD-MoE). To exploit temporal dynamics for continual adaptation, we decouple shared static components and temporal dynamics. During inference, temporal dynamics are used to assign dynamic weights to expert temporal encoders, enabling task-id-free adaptation without requiring task identifiers or stored exemplars.

3.1 Spatial and Spatiotemporal Encoder

Spatial Encoder. The shared adapter module Chen et al. (2022) $\mathcal{A}^b = \{\mathcal{A}_l^b\}_{l=1}^N$ is utilized with a frozen CLIP-ViT model with N layers of transformer module, serving as the spatial extractor. An adapter is an encoder-decoder architecture embedded into the residual of each transformer layer,

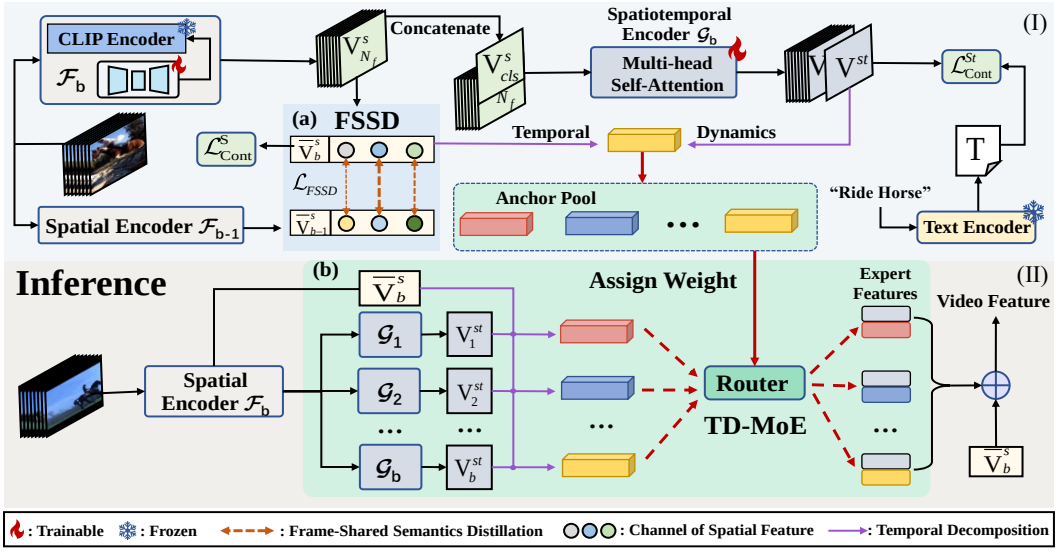

Figure 2: Overview of our proposed framework. (I) During training, the **(a) Frame-Shared Semantics Distillation (FSSD)** module retains past knowledge via frame-shared semantics importance-aware regularization. (II) At inference, the **(b) Temporal Decomposition-based Mixture-of-Experts (TD-MoE)** dynamically routes input videos to expert branches by leveraging the disentangled temporal component of the spatiotemporal representation, enabling adaptive predictions.

facilitating transfer learning and enhancing downstream task performance. Typically, it consists of a down-sampling MLP $\mathbf{W}_{down} \in \mathbb{R}^{d \times d_h}$, a ReLU activation $\phi(\cdot)$, and an up-sampling MLP $\mathbf{W}_{up} \in \mathbb{R}^{d_h \times d}$, where d is the input and output dimension, and d_h is the hidden dimension. For adapter, if input is $\mathbf{v}'_i \in \mathbb{R}^d$, which is the i-th frame-level feature after the Multi-Head Self-Attention and residual connection in CLIP ViT, the output of adapter is:

$$\mathbf{v}_i^a = \phi(\mathbf{v}'_i \mathbf{W}_{down})\mathbf{W}_{up}. \tag{1}$$

The spatial feature of i-th frame $\mathbf{V}_i^s = \mathcal{F}(V_i; \mathcal{A}^b), \mathbf{V}_i^s \in \mathbb{R}^{d_{vt}}$, where V_i is the i-th frame of the video, d_{vt} is the dimension of the aligned video-text features.

Spatiotemporal Encoder. To obtain the spatiotemporal representation, we feed both the frame-level spatial features and a learnable [CLS] token into a multi-head self-attention based spatiotemporal encoder $\mathcal{G}(\cdot)$. Specifically, the input to $\mathcal{G}$ consists of the frame features $\mathbf{V}_1^s, \ldots, \mathbf{V}_{N_f}^s$ and a [CLS] token $\mathbf{V}_{cls}^s \in \mathbb{R}^{d_{vt}}$, where N_f represents the number of sampled frames. The temporal encoder outputs the spatiotemporal feature $\mathbf{V}^{st}$, corresponding to the transformed [CLS] token:

$$\mathbf{V}^{st} = \mathcal{G}([\mathbf{V}_{cls}^s; \mathbf{V}_1^s; \ldots; \mathbf{V}_{N_f}^s])[0] \in \mathbb{R}^{d_{vt}}, \tag{2}$$

where $[0]$ selects the output associated with the [CLS] token after attention-based aggregation. More details for Spatial and Spatiotemporal Encoder are provided in appendix B.2 and B.3.

3.2 Frame-Shared Semantics Distillation

In VCIL, shared adapter modules inevitably drift in feature space when adapting to new tasks, leading to forgetting. Directly applying classic uniform-weighted distillation from CIL to video tasks ignores the differences in semantic importance and temporal stability across video features. This leads to a suboptimal balance between stability and plasticity. To address this, we propose Frame-Shared Semantics Distillation (FSSD), which identifies stable cross-frame channels capturing core semantics and regularizes them to preserve prior knowledge while allowing adaptation

Frame-Shared Semantics Distillation (FSSD). To mitigate semantic drift across tasks, we introduce a distillation loss weighted by the frame-shared semantics importance:

$$\mathcal{L}_{FSSD} = \frac{1}{|\mathcal{D}^b| \cdot d_{vt}} \sum_{c}^{|\mathcal{C}_b|} \sum_{i}^{N_c} \sum_{j}^{d_{vt}} I_{b-1,c,j} \cdot \|\bar{V}_{b-1,c,i,j}^s - \bar{V}_{b,c,i,j}^s\|_2^2, \tag{3}$$

where $|\mathcal{C}_b|$ is the number of classes in the b-th task, and N_c is the number of samples per class. $I_{b-1,c,j}$ denotes the frame-shared importance of the j-th channel for class c from task $(b-1)$. $\bar{V}_{b-1,c,i,j}^s$ and

$\bar{V}^s_{b,c,i,j}$ are the j-th channel outputs of the i-th sample in class c, extracted from the spatial encoders of task $(b-1)$ and b, respectively, calculated on current data, with the previous model frozen.

Frame-Shared Semantics. To quantify the importance of frame-shared semantics, we assess each channel based on two criteria: 1). **Semantic Sensitivity**. It measures the responsiveness to activation changes, thereby reflecting its reliability in representing consistent semantic information. and 2) **Classification Score**. It reflects the channel's contribution to the final classification.

For semantic sensitivity, we employ Fisher Information to estimate how sensitively a channel's activation influences the output. As spatial features $\bar{\mathbf{V}}^s = \frac{1}{N_f}\sum_{i=1}^{N_f}\mathbf{V}^s_i$ aggregate frame-wise variations, the Central Limit Theorem suggests each channel's distribution approximates a Gaussian (Belong to the same category). Thus, we assume the j-th channel activation for class c follows (For simplicity, we omit the subscripts for task and sample.):

$$\bar{V}^s_{c,j} \sim \mathcal{N}(\mu_{c,j}, \sigma^2_{c,j}), \tag{4}$$

where $\mu_{c,j}$ and $\sigma^2_{c,j}$ denote the mean and variance across frames. The Fisher Information $\mathcal{I}(\mu_{c,j})$ (Detailed derivations are provided in the appendix B.1) with respect to $\mu_{c,j}$ is computed as:

$$\mathcal{I}(\mu_{c,j}) = \mathbb{E}\left[\left(\frac{\bar{V}^s_{c,j}-\mu_{c,j}}{\sigma^2_{c,j}}\right)^2\right] = \frac{1}{\sigma^4_{c,j}}\mathbb{E}\left[(\bar{V}^s_{c,j}-\mu_{c,j})^2\right] = \frac{1}{\sigma^4_{c,j}}\cdot\sigma^2_{c,j} = \frac{1}{\sigma^2_{c,j}}. \tag{5}$$

For classification score, we compute the cosine similarity between the spatial video feature $\bar{\mathbf{V}}^s_c \in \mathbb{R}^{d_{vt}}$ and its corresponding text feature $\mathbf{T}_c \in \mathbb{R}^{d_{vt}}$. Specifically, for the j-th channel, the classification score is defined as $\gamma_{c,j} = \frac{\bar{V}^s_{c,j}\cdot T_{c,j}}{\|\bar{\mathbf{V}}^s_c\|\cdot\|\mathbf{T}_c\|}$, where $T_{c,j}$ denotes the j-th dimension feature of $\mathbf{T}_c$. We then take the expectation of $\gamma_{c,j}$ across frames to obtain a stable channel-level contribution estimate:

$$\mathbb{E}[\gamma_{c,j}] = \mathbb{E}\left[\frac{\bar{V}^s_{c,j}\cdot T_{c,j}}{\|\bar{\mathbf{V}}^s_c\|\cdot\|\mathbf{T}_c\|}\right] \propto \mathbb{E}\left[\frac{\bar{V}^s_{c,j}\cdot T_{c,j}}{\|\bar{\mathbf{V}}^s_c\|}\right] \approx \frac{T_{c,j}\cdot\mu_{c,j}}{\lambda}, \tag{6}$$

where $\|\bar{\mathbf{V}}^s_c\| \approx \lambda$ is treated as a constant after normalization. Combining semantic sensitivity and classification score, the semantic importance for the j-th channel of the c-th class is defined as:

$$I_{c,j} = \frac{T_{c,j}\cdot\mu_{c,j}}{\sigma^2_{c,j}}. \tag{7}$$

FSSD accumulates frame-shared semantic importance as distillation weights, retaining key channels for old tasks while allowing less important ones to adapt, thus balancing stability and plasticity.

3.3 Temporal Decomposition based Mixture-of-Experts

Given the high forgetting tendency of deep transformers in VCIL, we allocate a dedicated spatiotemporal encoder for each task. As task IDs are unavailable during inference, we allocate a spatiotemporal encoder per task and design a routing mechanism that dynamically weights experts based on temporal patterns, ensuring relevant experts contribute more to the final representation.

Task-Specific Expert. For each task, we train a dedicated expert based on the spatiotemporal encoder. The spatiotemporal features $\mathbf{V}^{st} \in \mathbb{R}^{d_{vt}}$ captured by each expert are computed as in Eq. 2.

Temporal Decomposition-based Router. To design this routing mechanism based on temporal dynamics, we consider two aspects: 1) **Temporal residuals**. These reflect the subtle temporal differences within redundant frames. 2) **Inter-frame information.** This captures abstract temporal concepts between frames, based on the knowledge learned by each expert.

For temporal residuals, we observe that redundant frames, where backgrounds and subjects remain consistent, cause minimal variation between adjacent frames Kim & Choi (2024); Liu et al. (2021). This leads to short-term temporal stationarity, which we further validate on the UCF101 and HMDB51 datasets Fig. 3. Thus, each frame feature is decomposed as $\mathbf{V}^s_i = \bar{\mathbf{v}} + \boldsymbol{\epsilon}_i$, with $\bar{\mathbf{v}}$ as shared static components and $\boldsymbol{\epsilon}_i$ as temporal residuals. The spatial representation is then the mean across frames:

$$\bar{\mathbf{V}}^s = \bar{\mathbf{v}} + \bar{\boldsymbol{\epsilon}}, \quad \bar{\boldsymbol{\epsilon}} = \frac{1}{N_f}\sum_{i=1}^{N_f}\boldsymbol{\epsilon}_i. \tag{8}$$

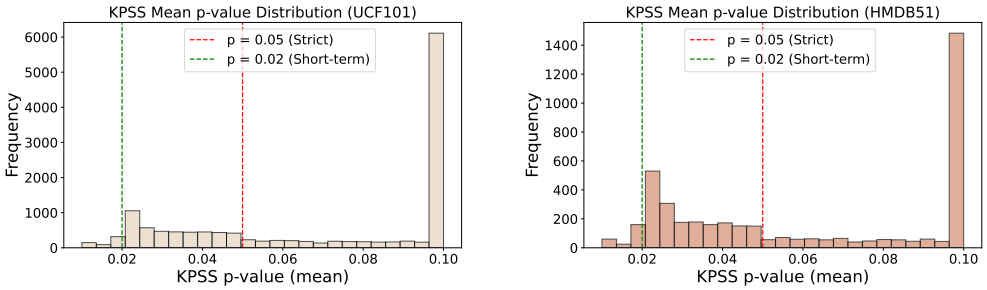

Figure 3: For each video, we uniformly sample 8 frames to compute the p-value defining $p > 0.05$ as strictly stationary and $p > 0.02$ as weakly stationary in the short term.

For the inter-frame information, since the spatiotemporal feature $\mathbf{V}^{st}$ is computed by the attention module, it can be approximated as $\mathbf{V}^{st} \approx \sum_{i=1}^{N_f} a_i \cdot \mathbf{V}_i^s$, where a_i is attention score. After normalization, we can obtain $\sum_{i=1}^{N_f} a_i = 1$. Substituting Eq. 8, we obtain:

$$\mathbf{V}^{st} = \bar{\mathbf{v}} + \sum_{i=1}^{N_f} a_i \cdot \boldsymbol{\epsilon}_i. \tag{9}$$

Since $\bar{\mathbf{v}}$ is difficult to estimate, and to decouple the temporal residual $\boldsymbol{\epsilon}_i$ and inter-frame information a_i, we naturally address this by using the difference between $\mathbf{V}^{st}$ and $\bar{\mathbf{V}}^s$, effectively isolating the temporal dynamics, which can be represented as:

$$\mathbf{V}^{tem} = \sum_{i=1}^{N_f} \left(a_i - \tfrac{1}{N_f} \right) \cdot \boldsymbol{\epsilon}_i. \tag{10}$$

This formulation reveals that $\mathbf{V}^{tem}$ quantifies the deviation between the model's attention-weighted temporal dynamics and the uniform temporal mean, effectively disentangling temporal variations from static semantics. This enables routing to exploit temporal cues while avoiding background interference, thereby mitigating forgetting and enhancing continual learning.

Inference. During inference, we first compute the decoupled temporal representation $\mathbf{V}^{tem} \in \mathbb{R}^{d_{vt}}$ for each input video. For all categories in the current task, we calculate the mean temporal representation and store it in the anchor pool as $\bar{\mathbf{V}}_c^{tem} \in \mathbb{R}^{d_{vt}}$, where c represents the c-th class. For each expert k, we compute a similarity-based score as the router:

$$r_k = \max_{c \in \mathcal{C}_k} \cos\left(\mathbf{V}_k^{tem}, \bar{\mathbf{V}}_c^{tem} \right), \tag{11}$$

where $\mathcal{C}_k$ represents the set of classes assigned to expert k. Then, we combine the adapter-tuned spatial features $\bar{\mathbf{V}}^s$ with the expert outputs weighted by r_k as the final video representation:

$$\mathbf{V} = \bar{\mathbf{V}}^s + \sum_k r_k \cdot \mathbf{V}_k^{st}. \tag{12}$$

The final video representation is matched with text embedding via cosine similarity for classification.

3.4 Loss Function and Optimization

Our loss function includes: 1) contrastive loss between video features and text descriptions for classification; 2) contrast loss between video features under adapter fine-tuning and text features for spatial optimization; and 3) FSSD loss to mitigate forgetting in shared adapter modules.

Contrastive Loss Formulation. We use symmetric contrastive loss for video-to-text and text-to-video alignment. Given a batch of N samples, let $\mathbf{V}_i$ and $\mathbf{T}_j$ denote the video and text features, respectively. The similarity between video i and text j is computed as the cosine similarity $S_{i,j} = \cos(\mathbf{V}_i, \mathbf{T}_j)$, forming a similarity matrix $S \in \mathbb{R}^{N \times N}$. Let $\mathbf{M} \in \{0,1\}^{N \times N}$ be the label mask, where $M_{i,j} = 1$ if $y_i = y_j$ and $M_{i,j} = 0$ otherwise. Then, the Video-to-text contrastive loss can be caculated by:

$$\mathcal{L}_{\text{v2t}} = -\frac{1}{N} \sum_{i=1}^{N} \log \left(\frac{\sum_{j=1}^{N} M_{i,j} \cdot \exp(S_{i,j})}{\sum_{j=1}^{N} \exp(S_{i,j}) + \varepsilon} \right), \tag{13}$$

Table 1: Average Accuracy ($\overline{\text{Acc}}$) of the UCF101 and HMDB51 under the TCD benchmark.

Method	Exemplar	Venue	UCF101			HMDB51	
			10×5s	5×10s	2×25s	5×5s	1×25s
iCaRL Rebuffi et al. (2017)	✓	CVPR'17	65.34	64.51	58.73	40.09	33.77
LwFMC Li & Hoiem (2017)	✗	TPAMI'18	42.14	25.59	11.68	26.82	16.49
LwM Dhar et al. (2019)	✗	CVPR'19	43.39	26.07	12.08	26.97	16.50
UCIR Hou et al. (2019)	✓	CVPR'19	74.09	70.50	64.00	46.53	37.15
PODNet Douillard et al. (2020)	✓	ECCV'20	74.37	73.75	71.87	48.78	46.62
TCD Park et al. (2021)	✓	ICCV'21	77.16	75.35	74.01	50.36	46.66
FrameMaker Pei et al. (2022)	✓	NeurIPS'22	78.64	78.14	77.49	51.12	47.37
L2P Wang et al. (2022c)	✗	CVPR'22	81.24	80.09	78.58	49.98	45.87
S-iPrompts Wang et al. (2022a)	✗	NeurIPS'22	80.60	80.27	80.43	53.11	53.89
ST-Prompt†Pei et al. (2023)	✗	CVPR'23	84.75	85.54	85.67	60.14	60.54
STSP Cheng et al. (2024)	✗	ECCV'24	81.15	82.84	79.25	56.99	49.19
HCE Liang et al. (2024)	✓	AAAI'24	80.01	78.81	77.62	52.01	48.94
StPR (Ours)	✗	–	**94.67**	**92.13**	**88.52**	**68.12**	**67.01**

Table 2: Average Accuracy ($\overline{\text{Acc}}$) of the SSv2 under the TCD benchmark, with best results in bold.

Method	Exemplar	Venue	**10** $\times$ **9**s	**5** $\times$ **18**s
iCaRL Rebuffi et al. (2017)	✓	CVPR'17	20.41	16.62
UCIR Hou et al. (2019)	✓	CVPR'19	24.32	19.31
PODNet Douillard et al. (2020)	✓	ECCV'20	27.63	20.14
TCD Park et al. (2021)	✓	ICCV'21	29.32	24.69
FrameMaker Pei et al. (2022)	✓	NeurIPS'22	31.41	26.57
L2P Wang et al. (2022c)	✗	CVPR'22	26.02	21.33
S-iPrompts Wang et al. (2022a)	✗	NeurIPS'22	33.69	30.84
ST-Prompt†Pei et al. (2023)	✗	CVPR'23	39.98	35.44
HCE Liang et al. (2024)	✓	AAAI'24	36.88	32.82
StPR (Ours)	✗	–	**40.79**	**37.30**

where ε is a small constant added to avoid division by zero. The Text-to-Video loss $\mathcal{L}_{\text{t2v}}$ is similarly defined by swapping video and text in the equation. Symmetric total contrastive loss is :

$$\mathcal{L}_{\text{Cont}} = \frac{1}{2}(\mathcal{L}_{\text{v2t}} + \mathcal{L}_{\text{t2v}}). \tag{14}$$

For $\mathcal{L}^{\text{St}}_{\text{Cont}}$, the embeddings are the spatiotemporal video feature $\mathbf{V}^{st}$ and corresponding text features. For $\mathcal{L}^{\text{S}}_{\text{Cont}}$, we use the CLIP adapter feature $\bar{\mathbf{V}}^{s}$ and corresponding text features.

Total Loss. The overall training loss is:

$$\mathcal{L} = \mathcal{L}^{\text{St}}_{\text{Cont}} + \mathcal{L}^{\text{S}}_{\text{Cont}} + w \cdot \mathcal{L}_{\text{FSSD}}, \tag{15}$$

where w is a hyperparameter. This design aligns both spatial and spatiotemporal semantics with text supervision, while the FSSD term preserves critical frame-shared semantics to mitigate forgetting.

4 Experiments

4.1 Experimental Details

Dataset. We evaluate our method on four benchmark datasets: UCF101 Soomro et al. (2012), HMDB51Kuehne et al. (2011), Something-Something V2 (SSv2) Goyal et al. (2017) and Kinetics400 Carreira & Zisserman (2017). All experiments are conducted in an exemplar-free setting. For fair comparison, we use the TCD benchmark Park et al. (2021) on UCF101, HMDB51, and SSv2, pretraining the model on 51, 26, and 84 base classes, respectively, with the remaining classes split into tasks. For Kinetics-400, we follow the vCLIMB benchmark Villa et al. (2022) with 10- or 20-task splits, each containing the same number of classes.

Evaluation Metrics. We adopt three widely-used metrics to evaluate performance in VCIL: 1). Final Accuracy (Acc) Villa et al. (2022), which measures the overall classification accuracy on all learned classes after the final task is completed; 2). Average Accuracy ($\overline{\text{Acc}}$) Park et al. (2021), which

Table 3: Results of the Kinetics-400 under the vCLIMB benchmark at 10 and 20 task settings.

Method	Exemplars	Venue	Kinetics400-10s		Kinetics400-20s	
			Acc ↑	BWF ↓	Acc ↑	BWF ↓
vCLIMB+BiC Villa et al. (2022)	✓	CVPR'22	27.90	51.96	23.06	58.97
vCLIMB+iCaRL Villa et al. (2022)	✓	CVPR'22	32.04	38.74	26.73	42.25
SMILE+BiC Alssum et al. (2023)	✓	CVPR'23	52.24	6.25	48.22	0.31
SMILE+iCaRL Alssum et al. (2023)	✓	CVPR'23	46.58	7.34	45.77	4.57
CSTA (Vivit) Chen et al. (2025)	✓	TCSVT'25	54.98	5.06	51.01	6.91
CSTA (Times) Chen et al. (2025)	✓	TCSVT'25	56.09	4.97	52.20	6.89
Ours	✗	–	**57.83**	14.01	**53.95**	15.09

Table 4: Ablation Study on UCF101 and HMDB51, with best results in bold.

Idx	$\mathcal{A}^b$	FSSD	TD-MoE	UCF101(5×10s)			HMDB51(5×5s)			UCF101(10×5s)			HMDB51(25×1s)		
				Acc ↑	$\overline{\text{Acc}}$ ↑	BWF ↓	Acc ↑	$\overline{\text{Acc}}$ ↑	BWF ↓	Acc ↑	$\overline{\text{Acc}}$ ↑	BWF ↓	Acc ↑	$\overline{\text{Acc}}$ ↑	BWF ↓
1	–	–	–	70.65	74.67	7.01	43.30	43.62	**6.18**	70.14	72.72	5.33	43.71	47.48	8.74
2	✓	✓	–	77.55	81.84	5.76	53.23	55.67	7.73	77.63	82.06	**4.86**	54.63	60.83	9.01
3	–	–	✓	79.33	89.36	12.38	56.12	61.14	11.75	85.94	93.47	7.40	62.54	68.88	10.20
4	✓	–	✓	83.07	91.28	10.52	57.47	63.37	21.30	88.03	94.14	8.39	64.71	73.02	21.72
5	✓	✓	✓	**85.79**	**92.13**	**5.63**	**63.04**	**68.12**	11.04	**88.85**	**94.67**	6.31	**69.61**	**75.07**	**7.02**

measures the mean classification accuracy over all incremental stages after the final task is completed; 3). Backward Forgetting (BWF) Villa et al. (2022), which quantifies the average drop in performance on previously learned tasks, reflecting how well the model retains past knowledge.

Implementation Details. All experiments are conducted on a single NVIDIA RTX 3090 GPU. We adopt the CLIP ViT-B/16 model Radford et al. (2021) as the backbone, with all its parameters frozen during training. The spatial and spatiotemporal encoders are the only trainable components in our framework. For optimization, we employ Stochastic Gradient Descent (SGD) with an initial learning rate of 0.01 and a batch size of 40. Each task is trained for 60 epochs in the first incremental session and 30 epochs in each subsequent session. The weighting hyperparameter w in Eq. 15 is set to 1×10^4. The multi-head self-attention module within the spatiotemporal encoder consists of three transformer layers, each employing two attention heads. Video clips are sampled using the TSN strategy Wang et al. (2018), selecting 8 frames per video uniformly across the temporal dimension.

4.2 Main Results

Table. 1, 2 and 3 report results on UCF101, HMDB51, SSv2 and Kinetics400, covering different action complexities and temporal dynamics. Based on their strategies to mitigate forgetting, existing methods are categorized into two groups: 1) Exemplar-based methods (iCaRL, UCIR, PODNet, TCD, FrameMaker, HCE, vCLIMB, SMILE, CSTA). They store video clips, frames, or compressed features and apply rehearsal to reduce forgetting. However, these methods face scalability and privacy challenges due to their reliance on stored exemplars. 2) CIL-based methods (LwFMC, LwM, L2P, S-iPrompts, ST-Prompt†, STSP). This group adapts techniques from image-based class-incremental learning, such as unified distillation and subspace projection, without storing exemplars. While avoiding exemplar storage, their performance tends to be lower, especially as task difficulty increases and lacking explainable spatiotemporal disentanglement. In contrast, Our method (StPR) without storing exemplars, surpasses all baselines across datasets and settings. On the TCD benchmark, our method outperforms the state-of-the-art approach (ST-Prompt†) as well as all exemplar-based methods on UCF101, HMDB51, and SSv2. On the vCLIMB benchmark, exemplar-based methods can alleviate forgetting by replaying stored samples, which makes forgetting lower. Nevertheless, our method achieves higher final accuracy, surpassing the current state-of-the-art (CSTA) and all exemplar-based counterparts.

4.3 Ablation Study

We perform ablation studies to evaluate the contribution of each component: the adapter tuning ($\mathcal{A}^b$), Frame-Shared Semantics Distillation (FSSD), and Temporal Decomposition-based Mixture-of-Experts (TD-MoE). Results are summarized in Table 4. The baseline (pretrained CLIP) model exhibits limited performance on downstream tasks, as it lacks adaptation to new task-specific categories. Introducing FSSD alone moderately improves performance by preserving spatial semantics and reducing semantic drift, while TD-MoE independently enhances adaptation by leveraging temporal

dynamics. However, using either module alone yields suboptimal performance. Combining adapter tuning with TD-MoE provides further improvements but still lacks sufficient stability in preserving spatial semantics. The full model (StPR), integrating both FSSD and TD-MoE, achieves the most stable performance across tasks, demonstrating the complementary strengths of spatial semantic preservation and temporal dynamic modeling.

4.4 Further analysis

Analysis of temporal-decomposition routing strategies. Table 5 compares our TD-MoE with several alternative Mixture-of-Experts (MoE) Jacobs et al. (1991) routing strategies. Simple averaging (Avg-MoE) and static weight assignments—including CLIP-MoE, which uses frozen CLIP visual features for routing, and Adapter-MoE, which uses adapter-tuned CLIP features—provide moderate improvements but fail to dynamically leverage task-specific temporal cues, often resulting in higher forgetting. In contrast, TD-MoE enables adaptive expert weighting based on temporal dynamics, consistently improving both accuracy and stability across tasks. This highlights the importance of modeling temporal variability explicitly, rather than relying on static or feature-agnostic routing.

Table 5: MoE Method Results on UCF101 and HMDB51, with best results in bold.

Method	UCF101(5×10s)		HMDB51(5×5s)		UCF101(10×5s)		HMDB51(25×1s)	
	Acc ↑	BWF ↓	Acc ↑	BWF ↓	Acc ↑	BWF ↓	Acc ↑	BWF ↓
Avg-MoE	81.14	9.95	59.46	8.49	84.04	9.60	62.69	9.14
CLIP-MoE	83.59	7.85	58.82	10.01	85.80	7.27	65.43	8.08
Adapter-MoE	83.26	6.07	61.99	**7.75**	84.31	7.82	65.57	7.68
TD-MoE(Ours)	**85.79**	**5.63**	**63.04**	11.04	**88.52**	**6.39**	**69.61**	**7.02**

Table 6: Distillation Method Results on UCF101 and HMDB51, with best results in bold.

Method	UCF101(5×10s)		HMDB51(5×5s)		UCF101(10×5s)		HMDB51(25×1s)	
	Acc ↑	BWF ↓	Acc ↑	BWF ↓	Acc ↑	BWF ↓	Acc ↑	BWF ↓
w/o Distillation	83.07	10.52	57.47	21.30	88.03	8.39	64.71	21.72
Distillation	84.27	7.45	61.74	13.33	88.12	7.95	67.54	13.38
FSSD(Ours)	**85.79**	**5.63**	**63.04**	**11.04**	**88.52**	**6.39**	**69.61**	**7.02**

Effectiveness of FSSD over Uniform Distillation. Table 6 compares our FSSD method with the no-distillation baseline (w/o Distillation) and standard uniform distillation (Distillation) across four VCIL settings. While uniform distillation improves accuracy and reduces backward forgetting (BWF) over the naive baseline, FSSD consistently outperforms both, achieving the highest accuracy and lowest BWF in all settings. These results highlight the benefit of selectively preserving frame-shared semantics, validating the importance-aware design of FSSD for continual video learning. For more experiments (such as **hyperparameter analysis** and visualization), see the appendix C

5 Conclusion

In this work, we propose StPR, a unified and exemplar-free framework for Video Class-Incremental Learning (VCIL) to tackle the spatiotemporal challenges in continual video learning. By disentangling spatial semantics and temporal dynamics, StPR effectively balances stability and plasticity without relying on stored exemplars. Our method combines Frame-Shared Semantics Distillation (FSSD), which selectively preserves meaningful and stable semantic channels, protecting model's plasticity. Temporal-Decomposition-based Mixture-of-Experts (TD-MoE), adaptively routes inputs based on temporal cues, reducing forgetting in deep networks. Extensive experiments on UCF101, HMDB51, and SSV2 validate the effectiveness and efficiency of our approach, establishing new state-of-the-art results for continual video recognition. In future work, we plan to explore more realistic application scenarios, such as open-world settings, and investigate the deployment of our method on resource-constrained edge devices. See Appendix D for reproducibility statement and Appendix E for our statement on LLM usage.

REFERENCES

Lama Alssum, Juan Leon Alcazar, Merey Ramazanova, Chen Zhao, and Bernard Ghanem. Just a glimpse: Rethinking temporal information for video continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2474–2483, 2023.

Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8218–8227, 2021.

Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021.

Matteo Boschini, Lorenzo Bonicelli, Angelo Porrello, Giovanni Bellitto, Matteo Pennisi, Simone Palazzo, Concetto Spampinato, and Simone Calderara. Transfer without forgetting. In *European Conference on Computer Vision*, pp. 692–709. Springer, 2022.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 532–547, 2018.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022.

Tieyuan Chen, Huabin Liu, Chern Hong Lim, John See, Xing Gao, Junhui Hou, and Weiyao Lin. Csta: Spatial-temporal causal adaptive learning for exemplar-free video class-incremental learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

Hao Cheng, Siyuan Yang, Chong Wang, Joey Tianyi Zhou, Alex C Kot, and Bihan Wen. Stsp: Spatial-temporal subspace projection for video class-incremental learning. In *European Conference on Computer Vision*, pp. 374–391. Springer, 2024.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021.

Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5138–5146, 2019.

Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Computer vision–ECCV 2020: 16th European conference, Glasgow, UK, August 23–28, 2020, proceedings, part XX 16*, pp. 86–102. Springer, 2020.

Beyza Ermis, Giovanni Zappella, Martin Wistuba, Aditya Rawal, and Cedric Archambeau. Memory efficient continual learning with transformers. *Advances in Neural Information Processing Systems*, 35:10629–10642, 2022.

Linxi Fan, Shyamal Buch, Guanzhi Wang, Ryan Cao, Yuke Zhu, Juan Carlos Niebles, and Li Fei-Fei. RubiksNet: learnable 3d-shift for efficient video action recognition. In *ECCV*, 2020.

Christoph Feichtenhofer. X3D: Expanding architectures for efficient video recognition. In *CVPR*, 2020.

Xinyuan Gao, Songlin Dong, Yuhang He, Qiang Wang, and Yihong Gong. Beyond prompt learning: Continual adapter for efficient rehearsal-free continual learning. In *European Conference on Computer Vision*, pp. 89–106. Springer, 2024.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 831–839, 2019.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

Hyun-Woo Kim and Yong-Suk Choi. Fusion attention for action recognition: Integrating sparse-dense and global attention for video action recognition. *Sensors (Basel, Switzerland)*, 24(21):6842, 2024.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114 (13):3521–3526, 2017.

Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pp. 2556–2563. IEEE, 2011.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Sen Liang, Kai Zhu, Wei Zhai, Zhiheng Liu, and Yang Cao. Hypercorrelation evolution for video class-incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 3315–3323, 2024.

Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23638–23647, 2024.

Ji Lin, Chuang Gan, and Song Han. TSM: Temporal shift module for efficient video understanding. In *ICCV*, 2019.

Xin Liu, Silvia L Pintea, Fatemeh Karimi Nejadasl, Olaf Booij, and Jan C Van Gemert. No frame left behind: Full video action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14892–14901, 2021.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 67–82, 2018.

Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost Van De Weijer. Class-incremental learning: survey and performance evaluation on image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5513–5533, 2022.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Jaeyoo Park, Minsoo Kang, and Bohyung Han. Class-incremental learning for action recognition in videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13698–13707, 2021.

Yixuan Pei, Zhiwu Qing, Jun Cen, Xiang Wang, Shiwei Zhang, Yaxiong Wang, Mingqian Tang, Nong Sang, and Xueming Qian. Learning a condensed frame for memory-efficient video class-incremental learning. *Advances in Neural Information Processing Systems*, 35:31002–31016, 2022.

Yixuan Pei, Zhiwu Qing, Shiwei Zhang, Xiang Wang, Yingya Zhang, Deli Zhao, and Xueming Qian. Space-time prompting for video class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11932–11942, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Roger Ratcliff. Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285, 1990.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International conference on machine learning*, pp. 4548–4557. PMLR, 2018.

James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. CODA-Prompt: COntinual Decomposed Attention-Based Prompting for Rehearsal-Free Continual Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11909–11919, June 2023. doi: 10.1109/CVPR52729.2023.01146.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Yuwen Tan, Qinhao Zhou, Xiang Xiang, Ke Wang, Yuchuan Wu, and Yongbin Li. Semantically-shifted incremental adapter-tuning is a continual vitransformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23252–23262, 2024.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

Andrés Villa, Kumail Alhamoud, Victor Escorcia, Fabian Caba, Juan León Alcázar, and Bernard Ghanem. vclimb: A novel video class incremental learning benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19035–19044, 2022.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.

Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018.

Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *Advances in Neural Information Processing Systems*, 36, 2024.

Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning. *Advances in Neural Information Processing Systems*, 35:5682–5695, 2022a.

Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*, pp. 631–648. Springer, 2022b.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 139–149, 2022c.

Xiang Xiang, Yuwen Tan, Qian Wan, Jing Ma, Alan Yuille, and Gregory D Hager. Coarse-to-fine incremental few-shot learning. In *European Conference on Computer Vision*, pp. 205–222. Springer, 2022.

Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23219–23230, 2024.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pp. 3987–3995. PMLR, 2017.

Dingwen Zhang, Yan Li, De Cheng, Nannan Wang, and Junwei Han. Center-sensitive kernel optimization for efficient on-device incremental learning. *arXiv preprint arXiv:2406.08830*, 2024.

Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19148–19158, 2023.

Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision*, pp. 1–21, 2024a.

Da-Wei Zhou, Hai-Long Sun, Han-Jia Ye, and De-Chuan Zhan. Expandable subspace ensemble for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23554–23564, 2024b.

Qinhao Zhou, Xiang Xiang, and Jing Ma. Hierarchical task-incremental learning with feature-space initialization inspired by neural collapse. *Neural Processing Letters*, pp. 1–17, 2023.

A APPENDIX: ALGORITHM

Algorithm 1: Frame-Shared Semantics Distillation (FSSD)

Input: Current task data $\mathcal{D}_b$; frozen model from task $b-1$; text features $\{\mathbf{T}_c\}$
Output: FSSD loss $\mathcal{L}_{\text{FSSD}}$

for *each class* $c \in \mathcal{Y}_b$ **do**
  Compute mean spatial features:
    $\bar{\mathbf{V}}^s_{b,c} = \frac{1}{N_f}\sum_{i=1}^{N_f} \mathbf{V}^s_{b,c,i}$; // Current model
    $\bar{\mathbf{V}}^s_{b-1,c} = \frac{1}{N_f}\sum_{i=1}^{N_f} \mathbf{V}^s_{b-1,c,i}$; // Previous model
  for *each channel* $j = 1$ *to* d **do**
    Estimate $\mu_{c,j}, \sigma^2_{c,j}$ from $\mathbf{V}^s_{b-1,c}$; // Across frames
    Compute semantic sensitivity:
      $\mathcal{I}(\mu_{c,j}) = \frac{1}{\sigma^2_{c,j}}$; // Fisher Information
    Compute classification contribution:
      $\mathbb{E}[\gamma_{c,j}] \propto \frac{T_{c,j}\cdot\mu_{c,j}}{\lambda}$; // Cosine-aligned score
    Compute importance score:
      $I_{b-1,c,j} = \frac{T_{c,j}\cdot\mu_{c,j}}{\sigma^2_{c,j}}$; // Weighted relevance
Compute weighted distillation loss:
$\mathcal{L}_{FSSD} = \frac{1}{|\mathcal{D}^b|\cdot d_{vt}}\sum_c^{|\mathcal{C}_b|}\sum_i^{N_c}\sum_j^{d_{vt}} I_{b-1,c,j}\cdot\|\bar{V}^s_{b-1,c,i,j} - \bar{V}^s_{b,c,i,j}\|_2^2$
return $\mathcal{L}_{FSSD}$

A.1 ALGORITHM OF TD-MOE

Algorithm 2: Temporal Decomposition based Mixture-of-Experts Inference

Input: Video frames $\{\mathbf{x}^{\text{V}}_i\}_{i=1}^{N_f}$;
  Task-specific experts $\{\mathcal{G}_k\}_{k=1}^{K}$;
  Temporal anchors $\{\bar{\mathbf{V}}^{tem}_c\}_{c=1}^{C}$ for current task
Output: Final representation $\mathbf{V}$

Compute spatial mean: $\bar{\mathbf{V}}^s = \frac{1}{N_f}\sum_{i=1}^{N_f}\mathbf{x}^{\text{V}}_i$; // Mean of frame features
for *each expert* $k = 1$ *to* K **do**
  Concatenate CLS token and frame features
  $\mathbf{V}^{st}_k = \mathcal{G}_k([\mathbf{x}^{\text{V}}_{\text{CLS}}; \mathbf{x}^{\text{V}}_1; \ldots; \mathbf{x}^{\text{V}}_{N_f}])[0]$; // Spatiotemporal feature
  Compute temporal representation:
  $\mathbf{V}^{tem}_k = \mathbf{V}^{st}_k - \bar{\mathbf{V}}^s$; // Temporal decomposition
  Compute routing score:
  $r_k = \max_{c\in\mathcal{Y}_k}\cos(\mathbf{V}^{tem}_k, \bar{\mathbf{V}}^{tem}_c)$; // Similarity to temporal anchors
Compute final representation:
$\mathbf{V} = \bar{\mathbf{V}}^s + \sum_{k=1}^{K} r_k\cdot\mathbf{V}^{st}_k$; // Residual fusion
return $\mathbf{V}$

Frame-Shared Semantics Distillation (FSSD). Algorithm 1 mitigates forgetting by selectively preserving spatial feature channels that are semantically important and temporally stable across frames. Importance is computed per channel using two criteria: (1) *Semantic sensitivity*, quantified by Fisher Information, and (2) *Classification contribution*, measured by cosine similarity with text features. These weights are used in a weighted distillation loss between the frozen previous model and the current task model, enabling exemplar-free knowledge retention while allowing plasticity.

Temporal Decomposition-based Mixture-of-Experts (TD-MoE). Algorithm 2 routes video inputs to task-specific experts based on temporal relevance. Each expert encodes spatiotemporal

features from video frames, from which temporal dynamics are isolated via residual decomposition. Temporal features are then compared to precomputed class anchors to compute routing scores. Final representations are generated by combining the expert outputs with the spatial feature via residual fusion. This enables dynamic, task ID-agnostic inference driven by temporal structure.

B Appendix: Theoretical Supplement

B.1 Frame-Shared Semantics

Semantic Sensitivity. The Fisher Information with respect to the mean parameter μ_j is defined as:

$$\mathcal{I}_j(\mu_{c,j}) = \mathbb{E}_{\bar{V}^s_{c,j}}\left[\left(\frac{\partial}{\partial \mu_{c,j}} \log p(\bar{V}^s_{c,j}; \mu_{c,j})\right)^2\right], \tag{16}$$

where $p(\bar{V}^s_{c,j}; \mu_{c,j})$ is the probability density function of the Gaussian:

$$p(\bar{V}^s_{c,j}; \mu_{c,j}) = \frac{1}{\sqrt{2\pi\sigma^2_{c,j}}} \exp\left(-\frac{(\bar{V}^s_{c,j} - \mu_{c,j})^2}{2\sigma^2_{c,j}}\right). \tag{17}$$

Taking the derivative of the log-likelihood with respect to μ_i:

$$\log p(\bar{V}^s_{c,j}; \mu_{c,j}) = -\frac{1}{2}\log(2\pi\sigma^2_{c,j}) - \frac{(\bar{V}^s_{c,j} - \mu_{c,j})^2}{2\sigma^2_i}, \tag{18}$$

$$\frac{\partial}{\partial \mu_{c,j}} \log p(\bar{V}^s_{c,j}; \mu_{c,j}) = \frac{\bar{V}^s_{c,j} - \mu_{c,j}}{\sigma^2_{c,j}}. \tag{19}$$

Then, the Fisher Information becomes:

$$\mathcal{I}(\mu_{c,j}) = \mathbb{E}\left[\left(\frac{\bar{V}^s_{c,j} - \mu_{c,j}}{\sigma^2_{c.j}}\right)^2\right] = \frac{1}{\sigma^4_j}\mathbb{E}\left[(\bar{V}^s_{c,j} - \mu_{c,j})^2\right] = \frac{1}{\sigma^4_j} \cdot \sigma^2_{c,j} = \frac{1}{\sigma^2_{c,j}}. \tag{20}$$

Thus, we obtain:

$$\mathcal{I}(\mu_{c,j}) = \frac{1}{\sigma^2_{c,j}}. \tag{21}$$

Classification Contribution. As we use cosine distance as classification basis, the c-th classification decision score $\gamma_{c,j}$ of the j-th channel is modeled as:

$$\gamma_{c,j} = \frac{\bar{V}^s_{c,j} \cdot T_{c,j}}{\|\bar{\boldsymbol{V}}^s_c\| \cdot \|\boldsymbol{T}_c\|} \propto \frac{\bar{V}^s_{c,j} \cdot T_{c,j}}{\|\bar{\boldsymbol{V}}^s_c\|}. \tag{22}$$

Therefore, the score function s can be approximately written as:

$$\gamma_{c,j} = \sum_j \bar{V}^s_{c,j} \cdot \alpha_j, \quad \alpha_j = \frac{T_{c,j}}{\|\bar{\boldsymbol{V}}^s_c\|} \approx \frac{T_{c,j}}{\lambda}. \tag{23}$$

For the expected score $\mathbb{E}[\gamma_{c,j}]$, it can be calculated as:

$$\mathbb{E}[\gamma_{c,j}] \propto \mathbb{E}\left[\frac{\sum_i \bar{V}^s_{c,j} \cdot T_{c,j}}{\|\bar{\mathbf{x}}_v\|}\right] = \sum_j \alpha_j \cdot \mu_i \approx \frac{T_{c,j} \cdot \mu_{c,j}}{\lambda} \tag{24}$$

Then, the joint measure of informativeness is:

$$I_{c,j} \propto \alpha_j \cdot \mu_{c,j} \cdot \mathcal{I}(\mu_{c,j}) \approx \frac{T_{c,j}}{\lambda} \cdot \frac{\mu_{c,j}}{\sigma^2_{c,j}}. \tag{25}$$

This expression provides a theoretically principled and interpretable metric for frame-shared semantics. It reflects the intuition that an informative channel should (i) be strongly activated on average ($\mu_{c,j}$ large), and (ii) exhibit consistent activation patterns across samples ($\sigma^2_{c,j}$ small). Therefore, we define the importance of channel j as:

$$I_{c,j} = \frac{T_{c,j} \cdot \mu_{c,j}}{\sigma^2_{c,j}}. \tag{26}$$

This formulation also aligns with the the signal-to-noise ratio theory (SNR), providing a unified theoretical justification.

B.2 SPATIAL ENCODER

We adopt a frozen CLIP-ViT model enhanced with shared adapters as the spatial encoder to extract frame-level features. Let a video $V = \{V_i\}_{i=1}^{N_f}$ consist of N_f uniformly sampled frames, where each frame V_i is processed independently. The spatial encoder $\mathcal{F}(V_i; \mathcal{A}^b)$ contains L transformer layers, each equipped with an adapter $\mathcal{A}_\ell^b$ inserted after the multi-head self-attention (MHSA) residual. The output of the encoder is a set of spatial features $\{\mathbf{v}_i^s\}_{i=1}^{N_f}$.

Transformer Block with Adapter. For the ℓ-th transformer layer, the feature update of input token $\mathbf{v}_i^{(\ell)} \in \mathbb{R}^d$ is computed as:

$$\mathbf{v}_i' = \mathbf{v}_i^{(\ell)} + \text{MHSA}(\mathbf{v}_i^{(\ell)}) \tag{27}$$

$$\mathbf{v}_i^a = \boldsymbol{\phi}(\mathbf{v}_i'^{\top} \mathbf{W}_{\text{down}}) \mathbf{W}_{\text{up}} \tag{28}$$

$$\mathbf{v}_i'' = \mathbf{v}_i' + \mathbf{v}_i^a \tag{29}$$

$$\mathbf{v}_i^{(\ell+1)} = \mathbf{v}_i'' + \text{FFN}(\mathbf{v}_i'') \tag{30}$$

$$\tilde{\mathbf{v}}_i^{(\ell+1)} = \mathbf{v}_i^{(\ell+1)} \mathbf{W}_{\text{proj}} \quad \text{(The last layer, projection to align text space)} \tag{31}$$

where $\boldsymbol{\phi}(\cdot)$ is a ReLU activation, $\mathbf{W}_{\text{down}} \in \mathbb{R}^{d \times d_h}$ and $\mathbf{W}_{\text{up}} \in \mathbb{R}^{d_h \times d}$ are the adapter projection weights, $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{d \times d_{vt}}$ is the projection weight that aligns the video and text feature spaces.

Multi-Head Self-Attention (MHSA). Given token sequence $\mathbf{Z} \in \mathbb{R}^{N \times d}$, multi-head self-attention is computed as:

$$\text{MHSA}(\mathbf{Z}) = \text{Concat}(\mathbf{h}_1, \ldots, \mathbf{h}_H) \mathbf{W}^O, \tag{32}$$

where each head $\mathbf{h}_h$ is computed as:

$$\mathbf{h}_h = \text{Softmax}\left(\frac{\mathbf{Q}_h \mathbf{K}_h^\top}{\sqrt{d_h}}\right) \mathbf{V}_h, \tag{33}$$

with projections:

$$\mathbf{Q}_h = \mathbf{Z}\mathbf{W}_h^Q, \quad \mathbf{K}_h = \mathbf{Z}\mathbf{W}_h^K, \quad \mathbf{V}_h = \mathbf{Z}\mathbf{W}_h^V, \tag{34}$$

and projection matrices $\mathbf{W}_h^Q, \mathbf{W}_h^K, \mathbf{W}_h^V \in \mathbb{R}^{d \times d_h}$, and $\mathbf{W}^O \in \mathbb{R}^{d \times d}$.

Feedforward Network (FFN). The FFN is a two-layer MLP with GELU activation:

$$\text{FFN}(\mathbf{v}) = \text{GELU}(\mathbf{v}\mathbf{W}_1)\mathbf{W}_2, \tag{35}$$

where $\mathbf{W}_1 \in \mathbb{R}^{d \times 4d}$ and $\mathbf{W}_2 \in \mathbb{R}^{4d \times d}$.

Output. After passing through L layers, the final spatial feature of the i-th frame is:

$$\mathbf{V}_i^s = \mathcal{F}(V_i; \mathcal{A}^b) \in \mathbb{R}^{d_{vt}}, \quad i = 1, \ldots, N_f. \tag{36}$$

These frame-level features $\{\mathbf{V}_i^s\}_{i=1}^{N_f}$ are then aggregated by the spatiotemporal encoder to form the global video representation.

B.3 SPATIOTEMPORAL ENCODER

Let a video clip be uniformly sampled into N_f frames. Each frame is processed by a spatial encoder to yield frame features $\mathbf{V}_i^s$. A learnable [CLS] token $\mathbf{V}_{\text{cls}}^s$ is prepended to the sequence, forming the input:

$$\tilde{\mathbf{V}}^s = [\mathbf{V}_{\text{cls}}^s; \mathbf{V}_1^s; \cdots; \mathbf{V}_{N_f}^s]. \tag{37}$$

This sequence is passed through a Transformer-based spatiotemporal encoder. Each layer comprises Multi-Head Self-Attention (MHSA), residual connections, and feedforward networks (FFNs). For a single attention head, the attention weights are computed as:

$$A_{ij} = \frac{\exp\left(\frac{\langle \mathbf{q}_i, \mathbf{k}_j \rangle}{\sqrt{d_k}}\right)}{\sum_{j'=1}^{N_f+1} \exp\left(\frac{\langle \mathbf{q}_i, \mathbf{k}_{j'} \rangle}{\sqrt{d_k}}\right)}, \tag{38}$$

where $\mathbf{q}_i = \mathbf{W}^Q \tilde{\mathbf{V}}_i^s$, $\mathbf{k}_j = \mathbf{W}^K \tilde{\mathbf{V}}_j^s$ and $\mathbf{v}_j = \mathbf{W}^V \tilde{\mathbf{V}}_j^s$ are the query, key, and value projections. The output of attention for token i is:

$$\mathbf{y}_i = \sum_{j=1}^{N_f+1} A_{i,j} \mathbf{v}_j. \tag{39}$$

We define the final spatiotemporal video representation as the output at the [CLS] token:

$$\mathbf{V}^{st} = \mathbf{y}_{\text{cls}} = \sum_{j=1}^{N_f+1} A_{\text{cls},j} \mathbf{v}_j. \tag{40}$$

This architecture enables spatiotemporal modeling by allowing: (i) long-range temporal interactions across frames via MHSA; (ii) spatial semantics retention from CLIP features; and (iii) adaptive fusion of information through the [CLS] token. Multiple attention heads further enhance expressiveness by learning diverse patterns. Consequently, $\mathbf{V}^{st}$ serves as a content-adaptive spatiotemporal descriptor capturing both motion and appearance.

C Appendix: Additional Experiments

C.1 Analysis of hyper-parameter.

Table 7: Analysis of hyperparameter w on HMDB51 and UCF101 datasets. Best results are in bold.

Dataset	Metric	1×10^3	1×10^4	2.5×10^4	5×10^4	1×10^5
HMDB(5×5s)	Acc ↑	59.51	63.04	62.10	62.29	**63.05**
	BWF ↓	16.90	**11.04**	11.14	12.32	12.77
UCF101(5×10s)	Acc ↑	85.10	**85.79**	84.54	84.88	84.16
	BWF ↓	7.93	**5.63**	6.95	5.67	6.83

Table. 7 analyzes the sensitivity of the hyper-parameter w, which controls the strength of FSSD regularization. Across a wide range of values, our framework maintains stable performance, indicating robustness to hyper-parameter variations. Moderate w values achieve the best trade-off between knowledge retention and adaptability, avoiding under-regularization or excessive constraint on model plasticity.

C.2 Complexity Analysis: FLOPs and Parameters

We compute the theoretical floating point operations (FLOPs) for each module: the CLIP ViT-B/16 backbone, the inserted adapters, and the spatiotemporal encoder module.

Table 8: FLOPs and parameter counts for each module per 8-frame video.

Module	Input Shape	Parameters	FLOPs (GFLOPs)
CLIP (ViT-B/16) (frozen)	$(8, 3, 224, 224)$	86M	269.81
Adapter	$(8, 197, 768)$	1.17M (1.36%)	3.73 (1.38%)
Spatiotemporal Encoder	$(9, 512)$	9.45M (10.99%)	0.0854 (0.03%)

Table 8 provides a detailed breakdown of computational complexity and parameter count for each component in our framework, evaluated on 8-frame video inputs. The backbone CLIP (ViT-B/16) dominates the overall cost with 86M parameters and 269.81 GFLOPs. The inserted adapter modules, despite being integrated into every transformer layer, introduce only 1.17M additional parameters (1.36%) and 1.38% more FLOPs, demonstrating their lightweight nature. Furthermore, our spatiotemporal encoder—used to capture dynamic information—adds merely 0.085 GFLOPs (0.03%, per-expert) and 9.45M parameters (10.99%), confirming its computational efficiency. These results validate that our method enhances temporal modeling with minimal overhead, making it well-suited for continual learning in resource-constrained settings.

C.3 Comparison of MoE Routing Strategies (task by task)

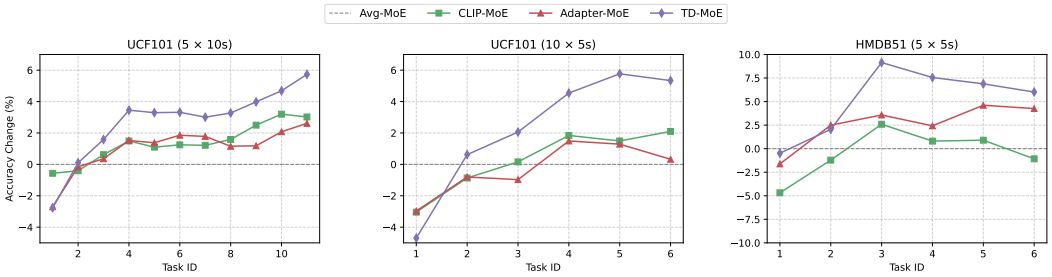

Figure 4: MoE Methods Performance per Task on UCF101 and HMDB51

Figure 4 presents the relative accuracy change (%) of different Mixture-of-Experts (MoE) routing strategies on three benchmarks: UCF101 with two task configurations (5×10s and 10×5s), and HMDB51 (5×5s). The horizontal axis represents the incremental task ID, while the vertical axis shows the accuracy change relative to the Avg-MoE baseline.

We observe that **TD-MoE consistently outperforms** all baselines across datasets and task granularities. Its performance advantage becomes more pronounced as the number of tasks increases, reaching up to 6–8% improvement on later tasks. In contrast, **Adapter-MoE and CLIP-MoE** exhibit only marginal gains, which tend to saturate early, suggesting limited ability to model task-specific dynamics. These findings confirm that temporal decomposition is effective for guiding expert selection in a task ID-agnostic manner, and helps mitigate forgetting by capturing relevant spatiotemporal cues.

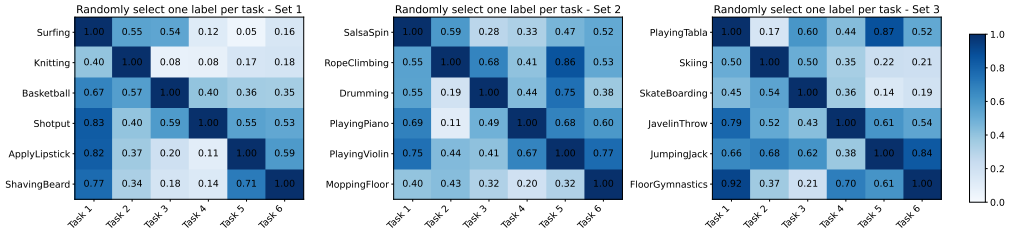

Figure 5: Heatmap of task selection by temporal decomposition-based router on the UCF101.

As shown in 5, temporal decomposition-based router is effective at task boundary decisions.

As shown in Figure 7, it maintains highest accuracy over time and effectively mitigates forgetting, demonstrating the complementary strengths of spatial semantic preservation and temporal dynamic modeling

C.4 Effectiveness of Frame-Shared Semantics Distillation (task by task)

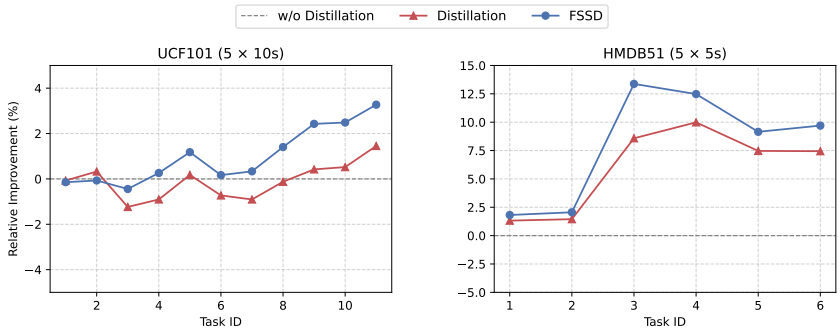

Figure 6: Distillation Methods Performance per Task on UCF101 and HMDB51

Figure 6 compares the impact of different distillation strategies, including w/o Distillation, unified Distillation, and our proposed Frame-Shared Semantics Distillation (FSSD), on UCF101 and HMDB51. The vertical axis reports the relative improvement over the non-distillation baseline.

The results demonstrate that **FSSD delivers the most consistent and significant improvements**, particularly on HMDB51. While unified distillation offers slight improvements, it lacks consistency across tasks. This suggests that uniform constraints fail to address the heterogeneous semantic importance of feature channels. Moreover, the increasing gap between FSSD and other methods over time confirms that **adaptive regularization based on frame-shared semantic importance is critical** for preserving relevant knowledge across tasks in VCIL.

C.5 Task-by-task Ablation Study Visualization

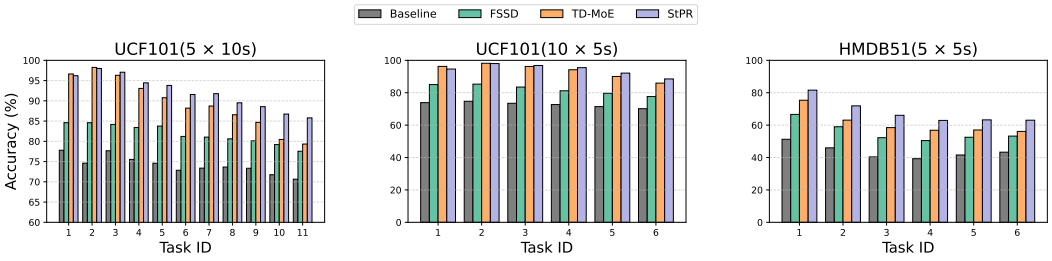

Figure 7: Task-wise ablation analysis across incremental tasks on UCF101 and HMDB51.

As shown in Figure 7, we progressively add our modules (FSSD and TD-MOE) on the UCF101 and HMDB51 datasets. Significant improvements are observed at each incremental stage, with more pronounced gains as the number of tasks increases, especially in the long-term scenario (10 tasks). This further validates the effectiveness of our proposed method and the superior performance of our model.

D Appendix: Reproducibility Statement

We detail the model and training setup in Sec.3, with datasets, preprocessing, and evaluation protocols in Sec.4. All hyperparameters and compute details are reported in Appx.C. Code is included in the supplementary materials.

E Appendix: LLM Usage Statement

In accordance with the ICLR 2026 policy on large language models (LLMs), we clarify that LLMs were employed solely to assist in polishing the language and improving readability of the manuscript. The conception of the research problem, development of the methodology, algorithmic design, code implementation, experimental setup, and result analysis were entirely carried out by the authors without reliance on LLMs.